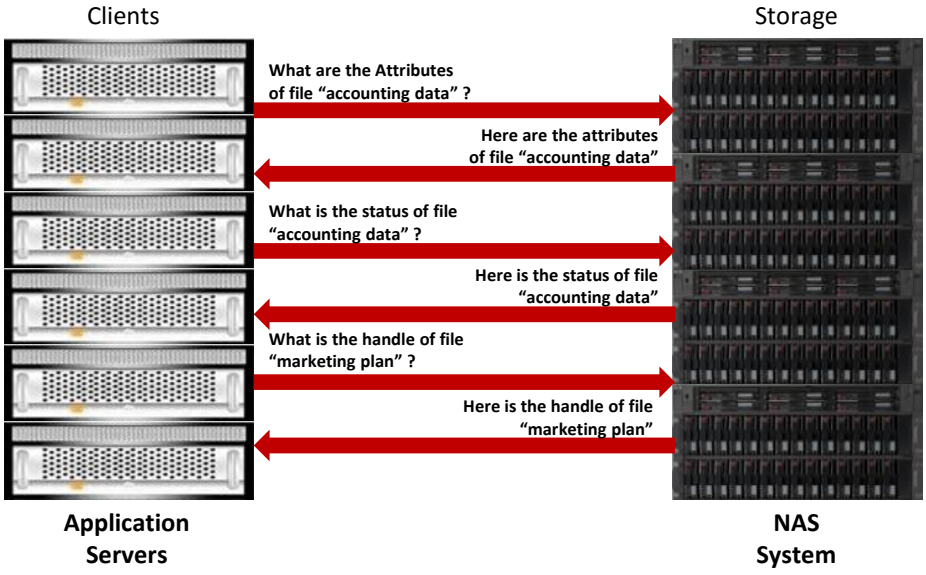
The Hidden Challenge to Network Attached Storage and Cloud Performance

The ever increasing rate of data growth puts a constant burden on existing storage systems to maintain performance and capacity goals. Storage administrators constantly need to modify and enhance storage infrastructure to meet application performance and data delivery goals. Frequently, fundamental read and write file access operations make up a surprisingly small percentage of the input/output (I/O) transactions that take place between clients and Network Attached Storage (NAS) systems. So what makes up the bulk of most I/O operations in a typical file workload? Data that describes data, which is called metadata.

When a client reads a file, browses a directory, or initiates a file write, several operations occur behind the scenes. These hidden operations give file systems like Network File System (NFS) and Server Message Block (SMB) information about the file, such as: when was the file last modified?, what is the file's pathname?, what is the size of the file?, who owns of the file?, who made the last file access?.

Metadata Requests Can Overwhelm NAS Systems



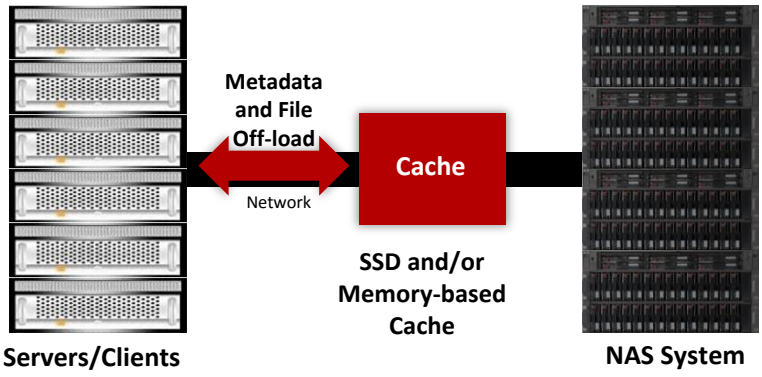
As much as 90% of all I/O transactions going to a NAS system are metadata requests.

In most workloads, the majority of all requests going to a NAS system will be metadata transactions, putting a large burden on the NAS system CPU even before a read or write action can take place.

Enhancing NAS System Performance with a Cache

Off-loading metadata request to a device that is faster than a NAS system is a win-win, since the off-loading device can focus on responding to the I/O intense metadata transactions, freeing up the NAS system to serve up bandwidth intensive read and write requests.

Caches Enhance NAS System Performance



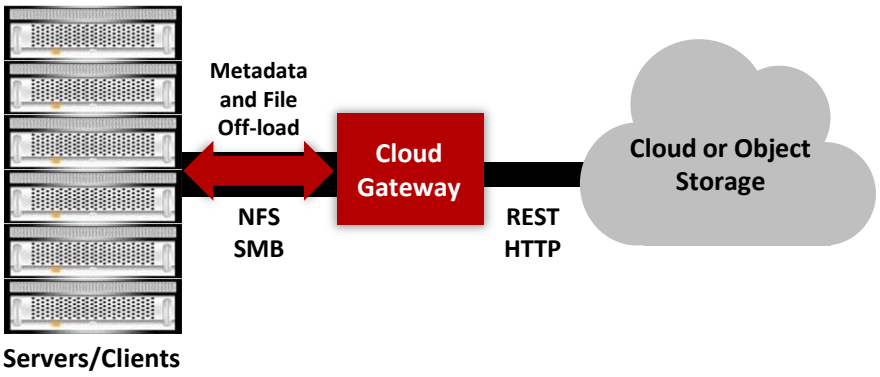
Caches performance-enhance data and metadata that are accessed frequently. Two of the challenges of caches is the "warm-up" time, and data changing its hot/cold state.

Caching systems have been used to off-load frequently accessed data that includes metadata and commonly accessed files. One of the problems with caching systems is they tend to be compute intensive, and do not scale well because they have to keep track of client/server sessions and file read/write/access counts. Caches also have a "warm-up" challenge: for a file or metadata to get copied into a cache, it typically needs to meet some hit count threshold. Since cache memory tends to be limited, and traffic patterns vary, actual performance benefits depend on the workload. In actual use, caches often fill up with mostly metadata.

The Performance Challenges of Using a Gateway to Make a Cloud Appear as a NAS System

Cloud Gateways are devices that act somewhat similar to a cache. They are installed in front of a cloud and translate cloud protocols like REST and HTTP into traditional file-based workflows. Specifically, cloud gateways attempt to make a cloud appear as a NAS system that supports the NFS and SMB file protocols.

Cloud Gateways Attempt To Make a Cloud Look and Perform Like A NAS System



Cloud gateways are challenged by the fact that they are trying to overcome a very large performance mismatch and therefore have unpredictable performance

Clouds are built for resiliency, not performance. They typically store data across geographically separate sites for disaster recovery. They are the perfect device to use low-cost, high-density storage due to their higher degree of resiliency compare to traditional storage systems and their ability to store massive amounts of data.

There is a huge challenge using a cloud in place of a traditional NAS system with a cloud gateway: the performance mismatch between what traditional NAS workloads expect and how fast a cloud gateway can respond can be significant.

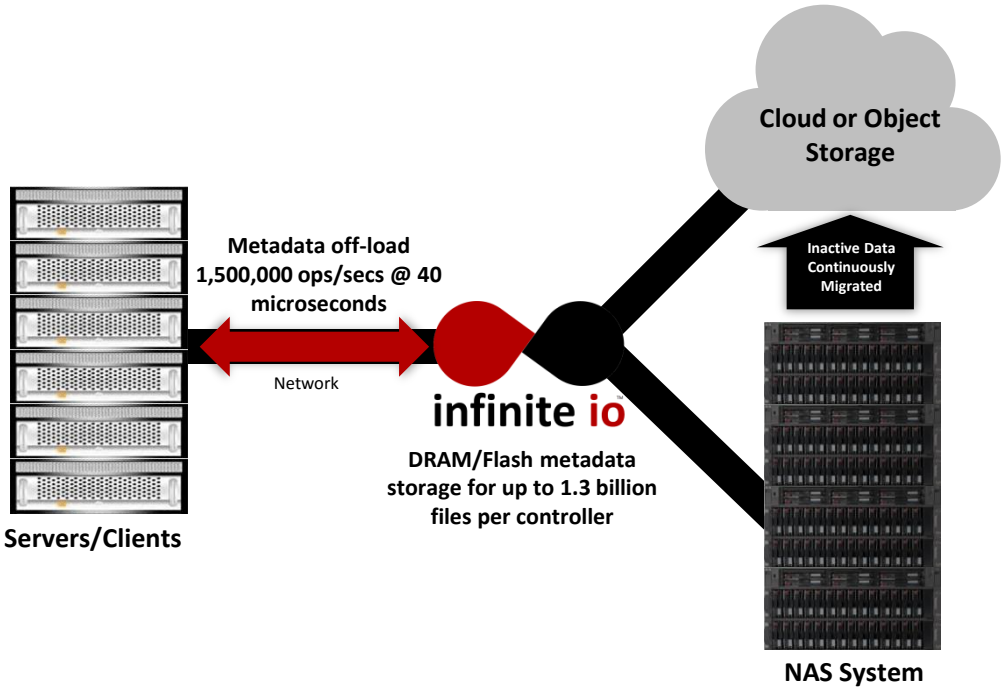
To try and overcome this major performance mismatch, cloud gateways operate similar to a cache by putting files and metadata into the gateway's storage as they are used or "warmed-up". The problem with this approach is while a specific file is "warming-up" it gets poor performance. Once "warmed-up", the file eventually "cools down", or the gateway's storage fills up, and the file is evicted from the gateway. If the file starts getting used again, the process repeats itself leading to unpredictable performance. Generally speaking, cloud gateways only work well in applications where performance is not a concern.

The Solution – Put All Metadata in Memory

infinite-io has taken an elegantly simple approach to solving the hidden cost of metadata requests with its network-based storage controllers — keep all metadata in DRAM and flash memory creating a metadata-map. It is not compute intensive, as there is no need to keep track of file hit counts. When first booted, the controllers scan the attached NAS systems and retrieve a copy of all metadata. There is no "warming-up" and there are no cache misses since all the metadata is stored in DRAM and flash

memory. Attached NAS systems are freed from responding to metadata requests and can focus on serving up actual file data.

The Metadata Map Enhances the Performance of Both Local and Cloud Storage



infinite-io takes a unique approach to metadata off-load by keeping a copy of all metadata for both local and cloud migrated files in memory, making the metadata-map always "hot"

Based on policies, the controllers continuously migrate inactive data to a cloud. Policy management tools give the insight to create policies that only migrate inactive data, significantly reducing cloud access. And if the cloud-migrated data is needed, most accesses will be metadata transactions that will be responded to directly out of memory by the controllers — effectively making a cloud or object store perform like a flash array for most workloads.